

Name: Satyam Rai
UID: 2019130051
TE COMPS
Batch C

Experiment 7

Aim: To design and implement an expert system, incorporating the match algorithm and the rule language.

Problem statement:

Read the below passage carefully and answer the questions:

Five cities all got more rain than usual this year. The five cities are Last Stand, Mile City, New Town, Olliopolis, and Polberg. The cities are located in five different areas of the country: the mountains, the forest, the coast, the desert, and in a valley.

The rainfall amounts were: 12 inches, 27 inches, 32 inches, 44 inches, and 65 inches.

- * The city in the desert got the least rain; the city in the forest got the most rain.
- * New Town is in the mountains.
- * Last Stand got more rain than Olliopolis.
- * Mile City got more rain than Polberg, but less rain than New Town.
- * Olliopolis got 44 inches of rain.
- * The city in the mountains got 32 inches of rain; the city on the coast got 27 inches of rain.

1. Which city got the most rain?
2. How much rain did Mile City get?
3. Which city is in the desert?
4. Where is Olliopolis located?

Code:

```
city(C) :-  
    length(C,5),  
    % CITY NAMES  
    member(h('Last Stand',_,_),C),  
    member(h('Mile City',_,_),C),  
    member(h('New Town',_,_),C),  
    member(h('Olliopolis',_,_),C),  
    member(h('Polberg',_,_),C),  
  
    % CITY REGIONS
```

```
member(h(_, 'mountains', _), C),
member(h(_, 'forest', _), C),
member(h(_, 'coast', _), C),
member(h(_, 'desert', _), C),
member(h(_, 'valley', _), C),
```

```
% RAINFALL AMOUNTS
```

```
member(h(_, _, 12), C),
member(h(_, _, 27), C),
member(h(_, _, 32), C),
member(h(_, _, 44), C),
member(h(_, _, 65), C),
```

```
% FACTS
```

```
% The city in the desert got the least rain;
% the city in the forest got the most rain.
member(h(_, 'desert', 12), C),
```

```
% New Town is in the mountains.
member(h('New Town', 'mountains', _), C),
```

```
% Last Stand got more rain than Olliopolis.
member(h('Last Stand', _, A), C),
member(h('Olliopolis', _, B), C),
A > B,
```

```
% Mile City got more rain than Polberg,
% but less rain than New Town.
member(h('Mile City', _, D), C),
member(h('Polberg', _, E), C),
member(h('New Town', _, F), C),
D > E,
D < F,
```

```
% Olliopolis got 44 inches of rain.
member(h('Olliopolis', _, 44), C),
```

```
% The city in the mountains got 32 inches of rain;
% the city on the coast got 27 inches of rain.
member(h(_, 'mountains', 32), C),
member(h(_, 'coast', 27), C).
```

```
get_city_rain(City, Rainfall):-
    city(C),
    member(h(City, _, Rainfall), C),
    write(City), write(" has received "), write(Rainfall), write(" inches"), nl.
```

```
get_city_region(City, Region):-  
    city(C),  
    member(h(City,Region,_),C),  
    write(City),write(" is in "),write(Region),write(" region"),nl.
```

Output:

The image displays four sequential screenshots of a Prolog interpreter window, each showing a query, its result, and a control bar with buttons for 'Next', '10', '100', '1,000', and 'Stop'.

- Query 1:** `get_city_rain(_,65).`
Result: Last Stand has received 65 inches
Output: true
- Query 2:** `get_city_rain('Mile City',_).`
Result: Mile City has received 27 inches
Output: true
- Query 3:** `get_city_region(_, 'desert').`
Result: Polberg is in desert region
Output: true
- Query 4:** `get_city_region('Olliopolis',_).`
Result: Olliopolis is in valley region
Output: true

Conclusion:

The given question has data about the amount of rainfall in a city and in which region the city lies in. In the above code, first all the names of cities and regions were stored which were mentioned in the problem statement and then the facts are stored.

I used prolog language in the above code as it helps to solve logic based questions easily by setting facts and rules for the question and then performing queries based on the rules and conditions mentioned. Then using the query in the above code, we can find the name of city from the amount of rainfall and vice-versa, also we can find the region from the city-name and vice-versa.